Bluetooth® LE SDK 4.2.1.0 GA
Gecko SDK Suite 4.1
October 19, 2022

Silicon Labs is a leading vendor in Bluetooth hardware and software technologies, used in products such as sports and fitness, consumer electronics, beacons, and smart home applications. The core SDK is an advanced Bluetooth 5.3 compliant stack that provides all of the core functionality along with multiple API to simplify development. The core functionality offers both standalone mode allowing a developer to create and run their application directly on the SoC, or in NCP mode allowing for the use of an external host MCU.

These release notes cover SDK version(s):

4.2.1.0 GA released October 19,2022 (early access part support plus fixed issues)
4.2.0.0 GA released September 28, 2022
4.1.0.0 GA released August 17,2022
4.0.0.0 GA released June 8, 2022

Bluetooth

KEY FEATURES

- Support added for Directed Advertising
- Support added for L2CAP connection - oriented channel
- Alpha Periodic Advertising Synchronization Transfer
- Apploader merged with Bootloader as communication plugin
- Alpha Dynamic Multiprotocol Bluetooth and multi-PAN 802.15.4 in RCP mode

Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Gecko Platform Release notes installed with this SDK or on the TECH DOCS tab on https://www.silabs.com/developers/bluetooth-low-energy. Silicon Labs also strongly recommends that you subscribe to Security Advisories for up-to-date information. For instructions as well as notes on using Secure Vault features, or if you are new to the Silicon Labs Bluetooth SDK, see Using This Release.

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 9.20.4.

- Using wine to build with the IarBuild.exe command line utility or IAR Embedded Workbench GUI on macOS or Linux could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.
- Customers on macOS or Linux are advised not to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

GCC (The GNU Compiler Collection) version 10.3-2021.10, provided with Simplicity Studio.

Contents

1 New Items

1.1 New Features

Added in release 4.2.0.0

New UC component bt_hci_coex "HCI Coex Vendor Specific Commands"

This component introduces several new vendor-specific HCI commands to control the coexistence feature.

Added in release 4.1.0.0

Improved configuration for bt_host_positioning example

The bt_host_positioning example can now accept individual configuration parameters via MQTT without the need to list the complete locator list.

Centralized runtime configuration of multi-locator setup

The broadcast configuration option is now in the same "silabs/aoa/config" MQTT topic and works for the bt_aoa_locator_host and bt_host_positioning examples.

Support of multiple connections in CBAP

Multiple connections are now supported between bt_soc_cbap central and peripheral devices.

Active scanner backoff procedure configuration

A new configuration option is added for active scanner, which is available through the Silicon Labs HCI vendor-specific command `CONFIG_KEY_ACTIVE_SCANNER_BACKOFF_UPPER_LIMIT`, that accepts an unsigned, 16 bit integer.

Configuration value is limited to 16 - 256, passing a value of 0 resets the maximum upper limit to the default (256).

Added in release 4.0.0.0

Apploader as Bootloader communication plugin on EFR32M|BG2x devices

Beginning with Bluetooth SDK version 4.0.x on EFR32M|BG2x devices, The AppLoader is merged with Gecko Bootloader as a communication plugin. The binary size of the combined functionality could be 16K smaller than the size of standalone Apploader and default Bootloader binaries. See AN1086: Using the Gecko Bootloader with Silicon Labs Bluetooth Applications for the details and how to migrate from old version of Bootloader and Apploader.

The Apploader remains as a standalone application on EFR32M|BG1x devices.

Directed Advertising feature

The Bluetooth stack supports legacy and extended directed advertising PDUs.

L2CAP Connection Oriented Channels feature

The Bluetooth stack supports L2CAP Connection-Oriented Channels in the LE Credit Based Flow Control mode.

New components for legacy and extended scanner

The Bluetooth stack has two new components: the bluetooth_feature_legacy_scanner for scanning legacy advertising PDUs, and the bluetooth_feature_extended_scanner for scanning legacy and extended advertising PDUs. They supersede some scanner functionality in the base bluetooth_feature_scanner feature.

If the application does not need to scan extended advertising PDUs, by using the bluetooth_feature_legacy_scanner and excluding the bluetooth_feature_extended_scanner, the number of received advertisement reports are reduced if devices advertising extended advertising PDUs are in the radio range. Another benefit is reduced application size by eliminating the stack functionalities for the bluetooth_feature_extended_scanner.

The sl_bt_scanner_set_timing() and sl_bt_scanner_set_mode() commands are not available to use if either the bluetooth_feature_legacy_scanner or bluetooth_feature_extended_scanner component is included. They are superseded by the sl_bt_scanner_set_parameters () command.

Alpha Periodic Advertising Synchronization Transfer (PAST)

The Periodic Advertising Synchronization Transfer (PAST) feature is provided as Alpha. New components and APIs include bluetooth_feature_past_receiver, bluetooth_feature_advertiser_past, bluetooth_feature_sync_past, and BGAPI classes "PAST Receiver", "Advertiser PAST" and "Sync PAST". The PAST feature is for evaluation purpose and the APIs are subject to change in a future release.

New option in command sl_bt_sm_configure

A new option is added in the sl_bt_sm_configure() command to require OOB data present from both devices when using secure connections OOB pairing.

Polarized antenna support in aoa_locator example

The AoA locator host example is extended with dual polarized antenna support.

New sample applications and components

New bt_soc_app_ota_dfu example is added to demonstrate background over-the-air (OTA) device-firmware-update (DFU) from the user application. New app_ota_dfu component is introduced for supporting freely adding the application OTA DFU to any component (e.g., bt_soc_empty).

Since the app_ota_dfu and in_place_ota_dfu (originally ota_dfu) components realize the same feature, only one can be selected by a project. Currently the app_ota_dfu component is an optional alternative to the in_place_ota_dfu.

New components ots_server_core, ots_client and ots_server are added for Object Transfer Service (Alpha).

New example is added for Certificate-Based Bluetooth Authentication and Pairing.

New example Certificate Signing Request Generator is added.

Custom vendor commands in RCP application

Custom HCI Command handlers are supported in RCP applications.

A new vendor-specific HCI command is added for specifying the PHYs that are allowed for use in extended scanning.

COEX signal identifier on EFR32M|BG24

The COEX signal identifier is supported on EFR32M|BG24.

1.2 New APIs

Added in release 4.2.0.0

Command sl_bt_coex_get_parameters: Get the current coexistence parameter settings.

Added in release 4.0.0.0

Command sl_bt_legacy_advertiser_start_directed: Start directed legacy advertising on an advertising set.

Command sl_bt_extended_advertiser_start_directed: Start directed extended advertising on an advertising set.

Event sl_bt_evt_scanner_legacy_advertisement_report: Reports an advertising data or scan response packet from an advertising device that uses legacy advertising PDUs. This event is used for reporting advertisements only if the application includes the bluetooth_feature_legacy_scanner or bluetooth_feature_extended_scanner component.

Event sl_bt_evt_scanner_extended_advertisement_report: Reports an advertising or scan response packet from an advertising device that uses extended advertising PDUs. This event is used for reporting advertisements only if the application includes the bluetooth_feature_extended_scanner component.

Command sl_bt_l2cap_open_le_channel: Create and configure an L2CAP channel on a Bluetooth connection using the LE credit-based connection request packet.

Command sl_bt_l2cap_send_le_channel_open_response: Send an LE credit based connection response to an LE credit-based connection request.

Command sl_bt_l2cap_channel_send_data: Send data to the peer channel endpoint on a Bluetooth connection.

Command sl_bt_l2cap_channel_send_credit: Send flow control credits to the peer channel endpoint indicating that the local channel endpoint is capable of receiving more data.

Command sl_bt_l2cap_close_channel: Send a disconnect request to close a credit-based logical channel.

Event sl_bt_evt_l2cap_le_channel_open_request: Indicates that an LE credit-based connection request on a Bluetooth connection is received.

Event sl_bt_evt_l2cap_le_channel_open_response: Indicates that an LE credit-based connection response is received.

Event sl_bt_evt_l2cap_channel_data: Indicates that data is received on a channel.

Event sl_bt_evt_l2cap_channel_credit: Indicates that flow control credits are received on a channel informing that the peer channel endpoint is capable of receiving more data.

Event sl_bt_evt_l2cap_channel_closed: Indicates that a credit-based logical channel is closed by the local or peer device.

Event sl_bt_evt_l2cap_command_rejected: Indicates that the peer device rejected a command.

Command sl_bt_connection_get_security_status: Get the security mode, encryption key size and bonding handle of a Bluetooth connection.

Command sl_bt_scanner_set_parameters: Provides the functionality of sl_bt_scanner_set_timing() and sl_bt_scanner_set_mode().

New option in command sl_bt_sm_configure: Bit 6 for configuring that OOB data from both devices is required when using secure connections OOB pairing.

2 Improvements

2.1 Changed Items

Changed in release 4.2.0.0

CBAP quality level changed to Alpha

In Bluetooth SDK v4.1.0, the 'Bluetooth - SoC Certificate Based Authentication and Pairing' and 'Bluetooth - SoC CSR Generator' example projects were mistakenly marked as production quality. In this release, the quality level of these example projects was adjusted to Alpha quality, as they did not yet reach production quality level.

Changed in release 4.1.0.0

Removed explicit EM2 requirement in the Bluetooth stack

The Bluetooth stack no longer explicitly requires EM2 mode at Bluetooth start. Instead, the stack relies on the lowest power mode configuration in the Power Manager component, which is EM2 by default. If the LF clock does not meet the accuracy requirement for specific Bluetooth functionalities, e.g., Bluetooth connections and periodic advertising etc., the Bluetooth stack will request EM1 mode at startup.

Changed in release 4.0.0.0

Assertion on Bluetooth stack initialization failure

An error returned by the Bluetooth stack initialization function sl_bt_stack_init is now asserted using the EFM_ASSERT provided by the sl_assert component.

Hardware revision in sl_bt_evt_system_boot event

The hardware revision information is reported in the boot event sl_bt_evt_system_boot. The major of the revision is in the the most-significant byte and the minor in the the least-significant byte.

Radio PA removed from the Bluetooth configuration

The radio PA configuration item has been removed from the Bluetooth configuration data structure sl_bt_configuration_t. The RAIL PA configuration is directly used by Bluetooth beginning with this release.

Code optimization in the Bluetooth stack and example applications

The Bluetooth stack has more code optimization in this release. Some example applications, such as bt_soc_empty, are optimized by using better-modularized software components. The size reduction could be from 2 to 7 KB depending on the device variants.

Renamed examples

All examples are renamed to follow new common naming convention.

Refactored multilocator angle correlation

Correlated angles component is refactored and renamed into a new, cleaner component: angle_queue.

Deviation value field in AoA host examples

Standard deviation value fields have been added to angle and position messages in the AoA host examples. Angle correction message format has been replaced with common angle message format.

New dependency in RCP HCI

In order to get a response to HCI_Le_Read_Phy, HCI_Le_Set_Default_Phy or HCI_Le_Set_Phy commands, the RCP-HCI application now requires the bluetooth_feature_connection_phy_update feature. This component conflicts with efr32M|BG1 devices because they only support LE 1M PHY.

Security Disabled on RCP_CPC

The security feature for CPC on RCP_CPC examples is disabled to save RAM.

To enable security, add the CPC SECURITY component..

2.2 Changed APIs

Changed in release 4.0.0.0

Command sl_bt_system_reset

On EFR32M|BG2x devices, the sl_bt_system_reset() command can only be used to boot to application mode. To boot to DFU mode, use the Bootloader API bootloader_rebootAndInstall().

Command sl_bt_scanner_stop

This command now returns SL_STATUS_OK if the scanner is not enabled. Previously error SL_STATUS_INVALID_STATE was returned.

Command sl_bt_ota_set_device_name

This command no longer supports EFR32M|BG2x devices as the result of Apploader merging with Bootloader.

Command sl_bt_ota_set_advertising_data

This command no longer supports EFR32M|BG2x devices as the result of Apploader merging with Bootloader.

Command sl_bt_ota_set_configuration

This command no longer supports EFR32M|BG2x devices as the result of Apploader merging with Bootloader.

Command sl_bt_ota_set_configuration

This command no longer supports EFR32M|BG2x devices as the result of Apploader merging with Bootloader.

3 Fixed Issues

Fixed in release 4.2.1.0

ID #	Description
1041997	Fix the librail_config libraries for the following xGM240 modules: BGM240PA22VNA, BGM240PA32VNA, BGM240PA32VNN, BGM240PB22VNA, BGM240PB32VNA, BGM240PB32VNN, MGM240PA22VNA, MGM240PA32VNA, MGM240PA32VNN. Without this update these modules will assert when trying to load the BLE PHYs.
1057775 1042933	Add BGM240P support in Bluetooth sample applications.

Fixed in release 4.2.0.0

ID #	Description
773951	Fix an issue in the 2M PHY CTE reception on 38.4 MHz clocked EFR32M|BG24 devices that caused high angle errors in the angle measurements.
816162	Fix an issue in the advertiser that almost doubled the advertising interval when the scanner task is also running.
834837	When the coexistence plugin is used, the AFH channel sweep no longer takes place when the grant signal is not set by the Wi-Fi device. This change is to avoid getting unreliable RSSI channel measurements while the Wi-Fi device is transmitting.
853176	Fix an issue in the Bluetooth Link Layer that caused the device not to enter EM2 in the duration between the AFH channel sweep and the previous advertising event.
1015746	Fix an issue in the Apploader library that caused OTA firmware update to not work on Series 2 modules.
1039558	Fix a bonding confirmation rejection issue when the local device, as a peripheral, already bonded with the peer peripheral device. Previously, there was a corner case when the bonding confirmation rejection was ignored. The device must be in the peripheral role and already bonded with the peer central device. Bonding confirmations must be enabled and the central device has deleted its bonding. If the peripheral device sends peripheral security request, the reply to bonding confirmation is ignored and devices will always carry out bonding.

Fixed in release 4.1.0.0

ID #	Description
756562	Fix an issue that the Bluetooth scanning on the LE Coded PHY on EFR32M|BG21 devices may hit a RAIL assertion (RAIL_ASSERT_FAILED_UNEXPECTED_STATE_RX_FIFO) causing the scanning to stop. The fix is available on Bluetooth SDK 4.0.0 onwards.
825839	Fix a timing issue that causes the Bluetooth Link Layer to lose relatively a higher number of packets and a higher PER when receiving DTM packets.
827415	The Bluetooth Link Layer now avoids blocking all the connection data channels when the AFH sweep decides that they are all jammed. Also, the cutoff threshold used to block the channels is now configurable through sl_bt_system_linklayer_configure() with key parameter = 18. Moreover, the blocked channels cooldown time is now configurable too through sl_bt_system_linklayer_configure() with key parameter = 19. This configuration is also available for the RCP application using HCI_VS_SiliconLabs_Configure command.
843142	Fix Bluetooth not working with MGM240L022RNF.
848767	The documentation of the Apploader component is labeled as "AppLoader application binary for EFR series 1 devices". However this component is also required by series 2 devices for reserving enough flash space for the new Bootloader that contains an Apploader library as the communication plugin.
849334	Fix an issue that would cause BGAPI commands on the NCP host to overflow the BGAPI command buffer if the data could not fit into a BGAPI command. The NCP host API now has checks for the payload length and command functions return SL_STATUS_COMMAND_TOO_LONG if command data is too long to fit.
851361	Fix an issue in the GATT server that, when using the dynamic GATT database feature, a service changed indication is sent over the active connection even if all the changes to the GATT DB were set in disabled state i.e., the GATT database has not changed from the GATT client's point of view.
855328	Fix an issue that causes the bt_host_positioning example not parsing the azimuth and elevation angle for all locators.
856669	Readme files for Example Projects are updated to reflect the changes in the bootloader compatibility. be sure to choose the proper bootloader for the Bluetooth application, according to the description.

ID #	Description
1018352	Fix an issue where DTM TX test started with sl_bt_test_dtm_tx_cw() or sl_bt_test_dtm_tx_v4() could use the wrong TX power.
1020307	Fix an issue where sl_bt_scanner_start() would use the PHY setting of a previous call and ignore the scanning_phy parameter in a later call if the application included the bluetooth_feature_legacy_scanner and/or bluetooth_feature_extended_scanner components.
1020891	Fix an issue where setting the CTUNE value using sl_bt_nvm_save(SL_BT_NVM_KEY_CTUNE, ...) would also affect other clock configuration settings. Starting from this version other clock configuration is now untouched and the CTUNE NVM key affects only the CTUNE value.

Fixed in release 4.0.0.0

ID #	Description
222747	There has been a feature request for the Apploader to be able to enter and stay in DFU mode for a certain time period on startup. On EFR32M|BG2x devices, this is now possible by customizing the functionality in file btl_apploader_common.c provided by the Apploader component.
271312 755025	There has been a feature request for the Apploader to support setting the TX power for OTA DFU. On EFR32M|BG2x devices, this is now possible by customizing the functionality in file btl_apploader_common.c provided by the Apploader component.
290592	There has been a feature request for the Apploader to support GPIO control in DFU mode. On EFR32M|BG2x devices, this is now possible by customizing the functionality in file btl_apploader_common.c provided by the Apploader component.
337467	There has been an issue that MGM12P has poor signal strength when doing OTA with Apploader. On EFR32M|BG2x devices, this is now possible to fix by enabling the FEM in the Bootloader and Apploader combined application.
645395	There has been an issue that the OTA DFU using Apploader fails on devices that do not use DCDC. On EFR32M|BG2x devices, this is now possible to fix by including the RAIL rail_util_pa component and configuring the item SL_RAIL_UTIL_PA_VOLTAGE_MV.
742632	Fix a memory leak in the Bluetooth bonding OOB mode on EFR32M|BG2x devices.
744709	Fix a scanning issue with BGM220P and BGM220S modules in RCP applications.
753951	Fix an issue that opening a connection to an advertisement using extended advertising PDUs sometimes does not succeed.
757736	The aoa_locator host example now tolerates the failure of a periodic advertising synchronization opening when the number of synchronizations has reached the configured maximum value in SL_BT_CONFIG_MAX_PERIODIC_ADVERTISING_SYNC.
757740	The aoa_locator host example now tolerates the failure of a Bluetooth connection opening when the number of connections has reached the configured maximum value in SL_BT_CONFIG_MAX_CONNECTIONS.
774799	Fix IAR compilation warnings of unreachable code in the simple_timer_freertos and simple_timer_freertos_static components.
815083	Fix the issue in coex commands that causes the request pin held high during the BLE scan dwell time.
829662 833981	Fix an issue that the coex BGAPI class is not added even if the RAIL Utility Coexistence component is installed. This issue caused unfunctional coex API calls at run time.
831505	Fix the issue that the advertiser only sends advertisement packets in one channel and does not respond to scan requests in the PTA mode.
835390	The Network Analyzer can now defragment Bluetooth extended advertisement packets correctly.
835393	The Network Analyzer can now detect Auxiliary Scan Responses correctly.
840603	Fix a memory allocation issue during parsing the antenna array configuration in the positioning host example.
842846	Fix an issue that the stack does not use the CE length values configured by the user application in responding to an L2CAP Connection Parameter Update request.
842924	Fix an issue on LE Coded PHY on EFR32M|BG21 devices that a corrupted coding indicator field on a received packet causes a timing calculation error which results in sync loss on the Bluetooth connection.

4 Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on https://www.si-labs.com/developers/bluetooth-low-energy in the Tech Docs tab.

ID #	Description	Workaround
361592	The sync_data event does not report TX power.	None
368403	If setting CTE interval to 1, a CTE request should be sent in every connection interval. But it is sent only in every second connection interval.	None
641122	The Bluetooth stack component does not provide a configuration for RF antenna path.	This is an issue specifically for BGM210P. One workaround is to manually update the configuration in sl_bluetooth_config.h in text edit mode. If the OTA with Apploader is used, include the bluetooth_feature_ota_config component in application project. Call command sl_bt_ota_set_rf_path() to set the RF path for OTA mode.
650079	LE 2M PHY on EFR32[B|M]G12 and EFR32[B|M]G13 doesn't work with smartphones using the Mediatek Helio chip due to an interoperability issue.	No workaround exists. For application development and testing, the disconnection can be avoided by disabling 2M PHY with sl_bt_connection_set_preferred_phy() or sl_bt_connection_set_default_preferred_phy().
682198	The Bluetooth stack has an interoperability issue on the 2M PHY with a Windows PC.	No workaround exists. For application development and testing, the disconnection can be avoided by disabling 2M PHY with sl_bt_connection_set_preferred_phy() or sl_bt_connection_set_default_preferred_phy().
730692	4-7% packet error rate is observed on EFR32M|BG13 devices when RSSI is between -25 and -10 dBm. The PER is nominal (as per the datasheet) both above and below this range.	None
756253	The RSSI value on a Bluetooth connection returned by the Bluetooth API is incorrect on EFR32M|B1, EFR32M|B12, EFR32M|B13, and EFR32M|B21 devices. On EFR32M|B21 devices. It is about 8~10 dBm higher than the actual value, according to a measurement.	Install the "RAIL Utility, RSSI" component in the application project. This component provides a default RSSI offset for the chip that is applied at the RAIL level and can help to achieve more accurate RSSI measurements.
845506	When the Bluetooth_feature_afh component for AFH is included, the feature initialization always enables AFH.	To include the component but not to enable AFH at device boot, change the parameter value from 1 to 0 in the function call of sl_btctrl_init_afh() in sl_bt_stack_init.c.
1019590	**Command sl_bt_coex_get_counters() always returns 0 for GRANT denied counts.**	**None**
1020072	**The coex REQUEST is held asserted for a longer duration than normal when a connection event pre-empts a lower-priority passive scan event that is at or near completion.**	**None**

5 Deprecated Items

Deprecated in release 4.0.0.0

Command sl_bt_advertiser_set_phy: deprecated and replaced by sl_bt_extended_advertiser_set_phy().

Command sl_bt_advertiser_set_configuration: deprecated and replaced by sl_bt_advertiser_configure().

Command sl_bt_advertiser_clear_configuration: Deprecated and replaced by sl_bt_advertiser_configure().

Command sl_bt_advertiser_set_data: Deprecated and replaced by sl_bt_legacy_advertiser_set_data() for legacy advertising PDUs, sl_bt_extended_advertiser_set_data() for extended advertising PDUs, and sl_bt_periodic_advertiser_set_data() for periodic advertising PDUs.

Command sl_bt_advertiser_set_long_data: Deprecated and replaced by sl_bt_extended_advertiser_set_long_data() for extended advertising PDUs and sl_bt_periodic_advertiser_set_long_data() for periodic advertising PDUs.

Command sl_bt_advertiser_start: Deprecated and replaced by sl_bt_legacy_advertiser_start() and sl_bt_extended_advertiser_start().

Command sl_bt_advertiser_start_periodic_advertising: Deprecated and replaced by sl_bt_periodic_advertiser_start().

Command sl_bt_advertiser_stop_periodic_advertising: Deprecated and replaced by sl_bt_periodic_advertiser_stop().

Commands sl_bt_scanner_set_timing and sl_bt_scanner_set_mode: This is a pre notice that the functionality of these two commands are replaced by new command sl_bt_scanner_set_parameters() and they will enter the deprecation notice period in the next major SDK release.

6 Removed Items

Removed from release 4.0.0.0

Example ncp_empty is removed, and replaced by bt_ncp example.

Removed APIs

The following APIs had been in deprecation notice period in the recent SDK versions, and the information of replacements can be found in v3.3 API reference manual.

Command sl_bt_sm_list_bonding_entry

Command sl_bt_sm_list_all_bondings

Command sl_bt_sm_set_oob_data

Command sl_bt_sm_use_sc_oob

Command sl_bt_sm_set_sc_remote_oob_data

Command sl_bt_system_set_soft_timer

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 4.1.0.0

851653 Added option to start cpcd conditional to a firmware version. Also allows firmware update process to be conditional to a firmware version. (-a/--app-version <version>). Added option to restart cpcd after firmware update. (-r/--restart-cpcd)

Added manufacturing library support on a Host-CMP RCP setup. This change allows RF testing on the manufacturing line on a Host-CMP RCP setup by sending mfglib EZSP frames from the Z3Gateway host to the RCP.

Added 802.15.4 CSL support to the multiprotocol RCP.

Added in release 4.0.0.0

A new concurrent multiprotocol configuration is available: Zigbee NCP and OpenThread RCP running concurrently on the EFR32, using the Co-Processor Communication (CPC) architecture. It is released as alpha quality. See *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP* for details.

7.2 Improvements

Changed in release 4.0.0.0

CPC security is now enabled by default in the cpcd.conf file and in the SLCP project files. This means data sent over the serial line between the host and the EFR32 is encrypted. A security commissioning step is required to bind the host to the EFR32. See https://github.com/SiliconLabs/cpc-daemon/blob/main/readme.md for details.

For convenience, the run.sh script in *app/host/multiprotocol/zigbeed/multiprotocol-container/* includes a `-K` argument for commissioning cpcd security when using the multiprotocol docker container.

The multiprotocol container has been upgraded to use ubuntu 22.04 and BlueZ 5.64.

The `zigbee_trust_center_backup` component now supports migrating from a Zigbee Host + NCP setup to a Zigbee Host + Zigbeed + RCP setup. See *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP* for details.

Zigbeed built from GSDK sources no longer requires the /accept_silabs_msla_file at runtime. Only the Zigbeed binary from the multiprotocol docker container requires it.

7.3 Fixed Issues

Fixed in release 4.2.0.0

ID #	Description
1025713	Increased the maximum length of the zigbeed device path from 40 to 4096.
1030557	Fixed build errors when adding the legacy bootloader interface component to the multiprotocol RCP projects.
1030557	Fixed a build issue to allow the legacy ember_bootloader_interface component to be added to MG1-based multiprotocol RCP projects, for customers using the legacy Ember bootloader. Also added support for CPCd to bootload devices with the legacy ezsp-spi Ember bootloader.

Fixed in release 4.1.0.0

ID #	Description
834191	Fixed cpc-hci-bridge so that it does not consume excessive CPU.
859224	Fixed an issue where CPC security was failing during initialization on MG1.
859301	Fixed an issue where Thread and Zigbee sleepy end devices failed to join to the RCP on MG1.
988216	Fixed issue where Zigbeed failed to run on a 64 bit Raspberry Pi.
824100	Fixed an issue where Z3GatewayCPC could not be started more than once without an error.
851331	Z3GatewayCpc can reconnect to cpcd service without needing a restart.
858153	Fixed a coex component configuration for MG1.
858503	Enabled the coex component by default in the multiprotocol RCP images, and fixed a RAM usage issue that prevented building on MG1.
1019947	Added support for building RCP projects for efr32mg1b andefr32mg1v parts.

Fixed in release 4.0.0.0

ID #	Description
760596	Reduced the CPU consumption of Zigbeed when it is idle.
811566	Fixed an issue where Zigbee sleepy end devices failed to join to an RCP parent.
817698	Fixed a Zigbeed crash due to a null buffer in the lower mac layer.
822233	Fixed an issue that caused dropped CPC packets over VCOM, especially at higher baud rates.
829614	The multi-PAN/multiprotocol 802.15.4 RCP now sets the radio tx power to the maximum of the power levels requested by all 15.4 host applications. This avoids the problem of one application reducing the power and inadvertently causing network connectivity problems for the other application.
830596	Fixed an issue where joining a sleepy end device to the multiprotocol RCP caused Z3Gateway to crash in some circumstances.
831689	Fixed an issue in Zigbeed that resulted in a fixed pan id and other parameters being chosen when forming a network.

7.4 Known Issues in the Current Release

ID #	Description	Workaround
811732	Custom token support is not available when using Zigbeed.	Support is planned in a future release.
828785	There is a known issue with the cpc-hci-bridge that causes the second HCI packet to be dropped if BlueZ sends two HCI packets to the RCP in rapid succession.	A fix is targeted for the next patch release.
829675	Bi-directional Green Power Devices can't pair with Z3GatewayGPCombo + Zigbeed + RCP.	A fix is being worked on for the next patch release.
937562	Bluetoothctl advertise on command fails with rcp-uart-802154-blehci app on Raspberry Pi OS 11.	Use Raspberry Pi OS 10 for running BlueZ.
1031607	**The rcp-uart-802154.slcp project is running low on RAM on an MG1 part. Adding components may reduce the heap size below what is needed to support ECDH binding in CPC.**	**A workaround is to disable CPC security via the SL_CPC_SECURITY_ENABLED configuration.**
1032183	**Zigbeed does not support EZSP coex commands.**	**Support is planned in a future release.**
1036622	**There is a problem using cmake to build ot-cli using the multi-PAN RCP. It will be fixed in a future release.**	**Use the make build target as described in AN1333 section 2.3.2.2.**

ID #	Description	Workaround
1040127	**CPC security fails to initialize for the rcp-uart-802154 and rcp-spi-802154 projects on mg13 and mg14 series parts.**	**Either disable CPC security, or add the mbedtls_entropy_adc component to the RCP image.**

7.5 Deprecated Items

None

7.6 Removed Items

Removed in release 4.0.0.0

Pre-built ARM binaries for multiprotocol host applications are no longer distributed within the GSDK (cpcd, otbr-agent, zigbeed, Z3Gateway, etc). These should be built from sources on the target platform using the instructions in *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP.*

A copy of sl_cpc.h that was being included in the OpenThread sources as a convenience has been removed. This header file is placed in the standard system location when cpcd is installed.

8 Using This Release

This release contains the following

- Silicon Labs Bluetooth stack library
- Bluetooth sample applications

For more information about the Bluetooth SDK see QSG169: Bluetooth® SDK v3.x Quick Start Guide. If you are new to Bluetooth see UG103.14: Bluetooth LE Fundamentals.

8.1 Installation and Use

The Bluetooth SDK is provided as part of the Gecko SDK (GSDK), the suite of Silicon Labs SDKs. To quickly get started with the GSDK, install Simplicity Studio 5, which will set up your development environment and walk you through GSDK installation. Simplicity Studio 5 includes everything needed for IoT product development with Silicon Labs devices, including a resource and project launcher, software configuration tools, full IDE with GNU toolchain, and analysis tools. Installation instructions are provided in the online Simplicity Studio 5 User's Guide.

Alternatively, Gecko SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/gecko_sdk for more information.

Simplicity Studio installs the GSDK by default in:

- (Windows): C:\Users\<NAME>\SimplicityStudio\SDKs\gecko_sdk
- (MacOS): /Users/<NAME>/SimplicityStudio/SDKs/gecko_sdk

Documentation specific to the SDK version is installed with the SDK. Additional information can often be found in the knowledge base articles (KBAs). API references and other information about this and earlier releases is available on https://docs.silabs.com/.

8.2 Security Information

Secure Vault Integration

When deployed to Secure Vault High devices, sensitive keys such as the Long Term Key (LTK) are protected using the Secure Vault Key Management functionality. The table below shows the protected keys and their storage protection characteristics.

Wrapped Key	Exportable / Non-Exportable	Notes
Remote Long Term Key (LTK)	Non-Exportable	
Local Long Term Key (legacy only)	Non-Exportable	
Remote Identity Resolving Key (IRK)	Exportable	Must be Exportable for future compatibility reasons
Local Identity Resolving Key	Exportable	Must be Exportable because the key is shared with other devices.

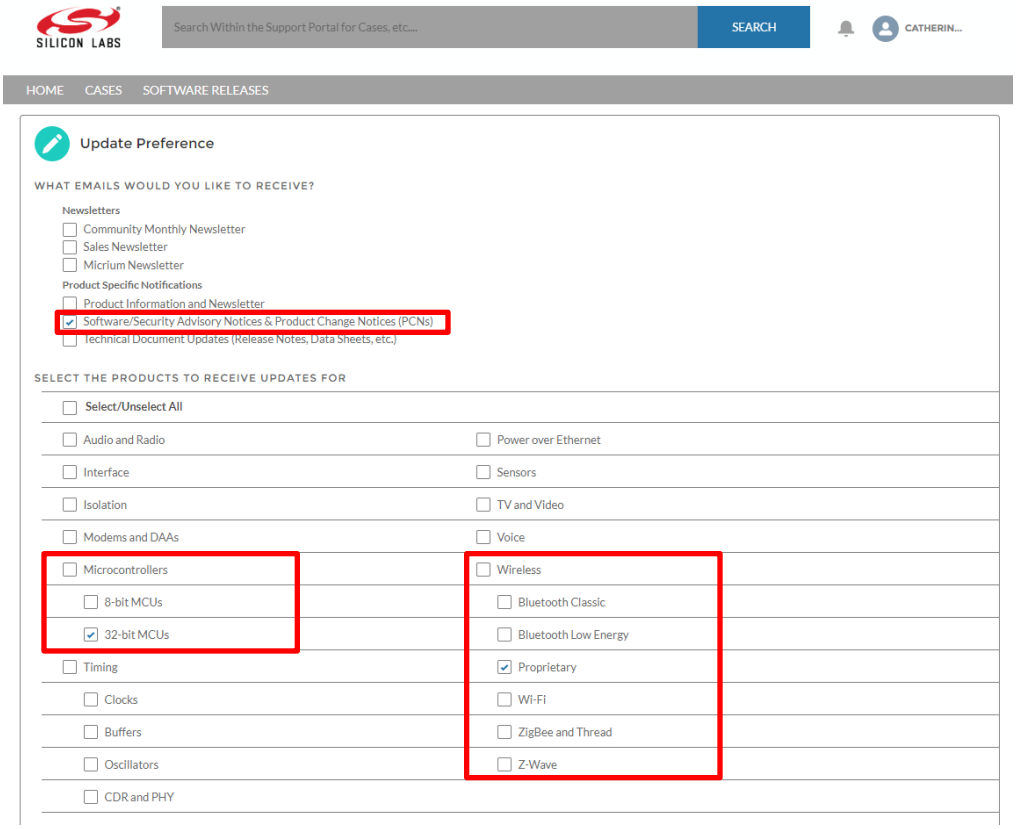
Wrapped keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.

Wrapped keys that are marked as "Exportable" can be used or shared at runtime but remain encrypted while stored in flash.

For more information on Secure Vault Key Management functionality, see AN1271: Secure Key Storage.

Security Advisories

To subscribe to Security Advisories, log in to the Silicon Labs customer portal, then select **Account Home**. Click **HOME** to go to the portal home page and then click the **Manage Notifications** tile. Make sure that 'Software/Security Advisory Notices & Product Change Notices (PCNs)' is checked, and that you are subscribed at minimum for your platform and protocol. Click **Save** to save any changes.



8.3 Support

Development Kit customers are eligible for training and technical support. Use the Silicon Labs Bluetooth LE web page to obtain information about all Silicon Labs Bluetooth products and services, and to sign up for product support.

You can contact Silicon Laboratories support at http://www.silabs.com/support.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com